

Greater Wisconsin Software Symposium

Sheraton Milwaukee Brookfield

Feb. 29 - Mar. 02, 2008

<http://www.nofluffjuststuff.com/conference/milwaukee/2008/02/index.html>

(event schedule as of March 2, 2008)

Fri, Feb. 29, 2008					
	Brookfield 1	Brookfield 2	Brookfield 3	Brookfield 4	Brookfield 5
12:00 - 1:00 PM	REGISTRATION				
1:00 - 1:15 PM	WELCOME				
1:15 - 2:45 PM	10 Tips for Getting Your Project Back on Track Jared Richardson	The Busy Java Developer's Guide to Debugging Ted Neward	JavaServer Faces: A Whirlwind Tour David Geary	Evolutionary SOA Neal Ford	Groovy, the Blue Pill: Writing Next Generation Java Code in Groovy Scott Davis
2:45 - 3:15 PM	BREAK				
3:15 - 4:45 PM	Gradual Agile: The Secret to Introducing Agile Practices Jared Richardson	The Busy Java Developer's Guide to Monitoring Ted Neward	Facelets David Geary	10 Ways to Improve Your Code Neal Ford	Groovy, The Red Pill: Metaprogramming, the Groovy Way to Blow a Buttoned-Down Java Developer's Mind Scott Davis
4:45 - 5:00 PM	BREAK				
5:00 - 6:30 PM	Credit Card Software Development: Recognizing and Repaying Technical Debt Jared Richardson	The Busy Java Developer's Guide to Performance and Scalability Ted Neward	Seam David Geary	Test Driven Design Neal Ford	Grails for (Recovering) Struts Developers: A Groovy Alternative Scott Davis
6:30 - 7:15 PM	DINNER - BROOKFIELD 4 & 5				
7:15 - 8:00 PM	Keynote: by Neal Ford				

Sat, Mar. 01, 2008					
	Brookfield 1	Brookfield 2	Brookfield 3	Brookfield 4	Brookfield 5
8:00 - 9:00 AM	BREAKFAST				
9:00 - 10:30 AM	"Design Patterns" in Dynamic Languages Neal Ford	Agile Software Testing Strategies Jared Richardson	The Busy Java Developer's Guide to Concurrency (Part 1: Threads) Ted Neward	Real World JSON Scott Davis	Know your Java? Venkat Subramaniam
10:30 - 11:00 AM	BREAK				
11:00 - 12:30 PM	BDD in Java and Groovy Venkat Subramaniam	Techniques 2008 Jared Richardson	The Busy Java Developer's Guide to Concurrency (Part 2: Concurrency) Ted Neward	YSlow: Building Your Website for Speed Scott Davis	Rich Faces David Geary
12:30 - 1:30 PM	LUNCH				
1:30 - 3:00 PM	The Busy Java Developer's Guide to Hacking (on) the JDK Ted Neward	Introduction to JRuby Neal Ford	Spring 2.5 - Spring without XML Ken Sipe	Design Patterns in Java and Groovy Venkat Subramaniam	Filthy Rich Clients with the Google Web Toolkit, Part I David Geary
3:00 - 3:15 PM	BREAK				
3:15 - 4:45 PM	DSL in Groovy Venkat Subramaniam	Rails for JRuby Neal Ford	Spring+JPA+Hibernate: Standards Meeting Productivity for Java Persistence Ken Sipe	10 Things Every Software Architect Should Know Richard Monson-Haefel	Filthy Rich Clients with the Google Web Toolkit, Part II David Geary
4:45 - 5:30 PM	BIRDS OF A FEATHER SESSIONS				

Sun, Mar. 02, 2008					
	Brookfield 1	Brookfield 2	Brookfield 3	Brookfield 4	Brookfield 5
8:00 - 9:00 AM	BREAKFAST				
9:00 - 10:30 AM	Acceptance Testing Application Behavior Venkat Subramaniam	Code Metrics & Analysis for Agile Projects Neal Ford	Java Memory, Performance and the Garbage Collector Ken Sipe	A Thorough Introduction To Groovy Jeff Brown	REST - Live! Brian Sletten
10:30 - 11:00 AM	BREAK				
11:00 - 12:30 PM	Caring about your Code Quality Venkat Subramaniam	Regular Expressions in Java Neal Ford	JMX and Spring: Manageability for Spring-based Applications Ken Sipe	Agile Test Driven Development With Groovy Jeff Brown	RESTlet for the Weary Brian Sletten
12:30 - 1:15 PM	LUNCH				
1:15 - 2:15 PM	EXPERT PANEL DISCUSSION				
2:15 - 3:45 PM	Groovy And Your Build Jeff Brown	Agile Project Management with Mingle Neal Ford	7 Habits of Highly Effective Developers Ken Sipe	Developing Rich Internet Applications Richard Monson-Haefel	Resource-Oriented Computing w/ NetKernel : Software for the 21st Century Brian Sletten
3:45 - 4:00 PM	BREAK				
4:00 - 5:30 PM	Grails - Agile Web 2.0 The Easy Way Jeff Brown	The Art of Producing Software: Applying Lean Concepts to Transform Your Software Development Organization John Carnell	Hacking - The Dark Arts Ken Sipe	Understanding Open Source Licensing Richard Monson-Haefel	The Semantic Web is Dead! Long Live the Semantic Web! Brian Sletten

Greater Wisconsin Software Symposium

Sheraton Milwaukee Brookfield

Feb. 29 - Mar. 02, 2008

<http://www.nofluffjuststuff.com/conference/milwaukee/2008/02/index.html>
(event schedule as of March 2, 2008)

REST - Live! by Brian Sletten

You've read the articles, the books, the Ph.D. thesis and all of the meta-commentary about building RESTful APIs, but you're still not sure where to begin. This is an interactive session with almost no slides. You should come prepared to pair program with me and everyone else in the room. We will tackle any reasonable suggestions of what might benefit from a RESTful interface. We will generate working code and explore the iterative process by which good REST-oriented APIs are created. Bring your ideas for open source projects that we might want to expose through a resource-oriented model.

RESTlet for the Weary by Brian Sletten

If you have started to take a look at REST as way of exposing web services or managing information spaces, you may be frustrated by the support offered by legacy containers. There is no direct support for REST concepts in the J2EE specs (yet). XML-based configurations are so 1990's. Come learn about Restlets, a little API that has caught the attention of many in the RESTafarian community.

Resource-Oriented Computing w/ NetKernel : Software for the 21st Century by Brian Sletten

Imagine the simplicity of REST married to the power of Unix pipes with the benefits of a loosely-coupled, logically-layered architecture. If that is hard to imagine, it may be because the architectures available to you today are convoluted accretions of mismatched technologies, languages, abstractions and data models. NetKernel is a disruptive technology that changes the game. It has been quietly gaining mind share in the past several years; people who are exposed to it don't want to go back to the tired and blue conventions of J2EE and .NET. Not only does it make building the kinds of systems you are building today easier, it does it more efficiently, with less code and a far more scalable runway to allow you to take advantage of the emerging multi-core, multi-CPU hardware that is coming our way. Come see how this open source / commercial product can change the way you think about building software.

The Semantic Web is Dead! Long Live the Semantic Web! by Brian Sletten

Just as the world is feeling comfortable with the Web, Tim Berners-Lee et al inform us that what we have seen so far is just the beginning. His original plans at CERN were larger and grander. The Semantic Web is a vision of machine-processable documents and metadata to improve search, knowledge discovery and data integration and management. The only problem is that there is no such thing. There is no Semantic Web, just the Web we have that is increasingly semantics-enabled. Forget the hype. Come learn how the technologies of this vision are being used today on the Web and in the Enterprise by more people than you might think.

JavaServer Faces: A Whirlwind Tour by David Geary

In April 2005, annual growth rates for jobs in JavaServer Faces, Struts, and Ruby on Rails were all at about 0%. Today, Struts' growth rate still hovers around 0%, but JSF and Rails have taken off. At the end of 2007, both JSF and Rails were growing at a rate of between 400-500% annually (according to indeed.com). JSF has passed the adoption tipping point, and is now the Java-based framework of choice, as is evidenced by its ecosystem. From vendors such as MyEclipse and RedHat to open source projects such as Seam, Facelets, and Ajax4JSF, JSF is where the action is. Come see why JSF is so popular. In this code- and demo-intensive session, I'll show you the fundamentals of JSF. **Prerequisite:** *Some knowledge of Java-based web applications, such as Struts, is a plus, but is not required. If you have a significant experience with JSF, you probably already know most of what's covered in this session.*

Facelets by David Geary

Facelets is a combination of Tiles and Tapestry, and it's the hottest JSF-related open source project on the planet. It's popularity is well deserved, and in fact, much of what is in Facelets today will make its way into the JSF 2.0 spec due out in 2008. So not only can you come to this session and see some really cool demos that you can put to use in the real world, but you'll also be learning JSF 2.0 before it's even been defined! How's that for a ROI? **Prerequisite:** *Some knowledge of JSF is essential. If you're familiar with a templating framework, such as Velocity or Tiles, that's a plus, but not required.*

Seam by David Geary

Have you ever stopped to think that you need to learn two frameworks to develop a non-trivial, database-backed, web application? Struts and iBatis; JSF and Hibernate; Tapestry and EJB3.0. Two frameworks. And then you have to learn to use them together. Why do we have to learn two frameworks just to retrieve "Hello World" from a database and show it in a view. Isn't that crazy? Now you can use one framework, and use one component model. One. Isn't that nice? Seam, a framework built on JSF and EJB3.0, unifies the JSF and EJB component models. Seam is a steam roller, quickly gathering market share among JSF newbies and longtime believers alike. Come see what it's all about. **Prerequisite:** *Some knowledge of JSF is required. If you don't know what a managed bean is, for instance, then attend JSF Whirlwind before this session.*

Rich Faces by David Geary

This talk explores the RichFaces Ajax framework, which is really two frameworks: Ajax4jsf and RichFaces components. In this session you will see how to implement low-level Ajax functionality using Ajax4JSF, and how to use high-level Ajax components from RichFaces. **Prerequisite:** *Some knowledge of JSF is required, in addition to familiarity with Ajax.*

Filthy Rich Clients with the Google Web Toolkit, Part I by David Geary

The Google Web Toolkit (GWT) is truly a revolutionary framework that lets you develop Ajaxified web applications without knowing anything about Ajax or JavaScript. But the GWT goes way beyond basic Ajax by letting you implement desktop-like applications that run in the ubiquitous browser.

Filthy Rich Clients with the Google Web Toolkit, Part II by David Geary

In the second part of this talk, you will learn how to extend the GWT by implementing custom widgets, including a scrolling viewport and a drag and drop framework. After discussing custom widgets, you will see how to integrate database access into your GWT applications, and how to deploy your GWT applications to external servers.

10 Tips for Getting Your Project Back on Track by Jared Richardson

Software projects fail over and over for many of the same reasons. We'll look at some of the more avoidable problems and some solid ways to fix them, or avoid them in the first place.

Gradual Agile: The Secret to Introducing Agile Practices by Jared Richardson

Agile practices are popular because they work, but getting people to take that first step can be tricky.

Credit Card Software Development: Recognizing and Repaying Technical Debt by Jared Richardson

Technical debt has long been recognized in technical circles for years, but convincing your manager to budget time to repay "technical debt" has always been problematic. Let's couch the term technical debt concept in language more familiar to our managers: credit card debt.

Agile Software Testing Strategies by Jared Richardson

Creating and maintaining a solid automated test suite is critical to an Agile strategy, but often we're just told to "Do it." In this talk we'll look at several pragmatic strategies for creating and building your suite.

Techniques 2008 by Jared Richardson

There are a number of great techniques you can use across technologies and projects. Come hear some of my favorites and contribute a few of your own. We'll discuss topics from DRY to creating a zone defense for your product.

A Thorough Introduction To Groovy by Jeff Brown

Groovy is an agile dynamic language for the Java platform. The language and its libraries bring many things to the table to ease the process of building applications for the Java platform. This session provides a detailed run through Groovy with lots of code samples to drive home the power of the language.

Agile Test Driven Development With Groovy by Jeff Brown

Dynamic languages bring a lot of interesting elements to the table for teams interested in doing Test Driven Development (TDD). Groovy lends itself very well to TDD and this session demonstrates many features of the language and its libraries that help teams build more testable systems and build better tests.

Groovy And Your Build by Jeff Brown

There are numerous roles that Groovy may play in your build process to greatly simplify the management of the build while bringing more capabilities. This session will detail a lot of the things that Groovy can do to improve your build and lessen the amount of effort you spend on your build. **Prerequisite:** *A Thorough Introduction To Groovy*

Grails - Agile Web 2.0 The Easy Way by Jeff Brown

Grails is a full stack MVC framework for building web applications for the Java platform. Grails makes web application development both fun and easy. This session covers all of the fundamentals of building web applications with Grails.

The Art of Producing Software: Applying Lean Concepts to Transform Your Software Development Organization by John Carnell

Waste is an insidious beast that drains the productivity of development teams and the organizations they work in. Many organizations are now realizing that by turning their gaze inward they can streamline their overall development processes, deliver higher quality products faster and save significant amounts of money. In this talk we will look at the "Lean" techniques first developed by companies like Toyota and how they can be applied to common software development practices. We will walk through such concepts as identifying the different types of waste you might encounter in a software development effort, using Value Stream Mapping (VSM) to help measure the impact of that waste and different techniques you can use to eliminate that waste.

Spring 2.5 - Spring without XML by Ken Sipe

Spring 2.5 is brand spanking new, with a number of fantastic features. With growth of large and complex Spring applications which struggle with xml manageability and with the added pressure of Guice and SEAM there is a push for less XML, with solution leaning towards annotations. Spring 2.5 adds to the toolset provided in Spring 2.0 to provide a development environment where XML is greatly reduced... or eliminated if you so choose.

Spring+JPA+Hibernate: Standards Meeting Productivity for Java Persistence by Ken Sipe

Well the standards created EntityBeans.... yea. and the community created Hibernate. Fortunately the standards body learned some lessons and created JPA. JPA requires a vendor implementation and none make a better choice then Hibernate. Combined with Spring this trio is a powerhouse when it comes to developer productivity on applications requiring persistence.

Java Memory, Performance and the Garbage Collector by Ken Sipe

You are using Java, whew!!! No need to worry about memory, the garbage collector will handle that. Those who have had a memory issue in Java are not so naive any more. Often memory utilization and heap sizes are an after thought and are not recognized until the application is in production, often caused by application uptime, production request volume or production sets of data. When the OutOfMemory Error occurs, often the science of development seems to brake down and knobs are turned. First the (-mx) maximum heap space gets adjusted... More is better right. The next OutOfMemory, heads start scratching, code reviews start in earnest, and Google gets several new hits. Did you know that it is possible to get an OutOfMemory error without running out of heap space?

JMX and Spring: Manageability for Spring-based Applications by Ken Sipe

This session describes management of Java resources using the Java Management Extensions JMX API. JMX provides a unified framework to instrument Java systems with monitoring and management capabilities.

7 Habits of Highly Effective Developers by Ken Sipe

Thoughts lead to words, words lead to action, actions lead to habits. In this session we'll sharpen the development saw in the process of understanding what makes a hyper-productive programmer. The focus will consist of developer habits and development processes.

Hacking - The Dark Arts by Ken Sipe

A live Hacking demonstration exposing the tools and techniques used by Hackers.

Evolutionary SOA by Neal Ford

This session demonstrates that "Agility" and "SOA" complement each other quite well. Just because SOA is buzz-word compliant doesn't mean that you should throw good practices out the window. This session demonstrates how you can apply the principles of agility to building highly complex distributed enterprises.

10 Ways to Improve Your Code by Neal Ford

No one writes perfect code: even the best developers fall into bad habits and traps. This talk illustrates blind spots and helps you write better code.

Test Driven Design by Neal Ford

Most developers think that "TDD" stands for Test-driven Development. But it really should stand for "Test-driven Design". Rigorously using TDD makes your code much better in multiple ways.

Keynote: Ancient Philosophers & Blowhard Jamborees by Neal Ford

It turns out that ancient philosophers knew a lot about software -- did you know that Plato defined object-oriented programming? This keynote applies old lessons to new problems and old problems to new lessons. It describes why SOA is so hard, and why people in your company make bone-headed decisions. What other keynote includes Rube Goldberg, Aristotle, Dave Thomas, and India?

"Design Patterns" in Dynamic Languages by Neal Ford

The Gang of Four book should have been entitled "Palliatives for Statically Typed Languages", because the recipes it provides are cumbersome solutions to the problems it poses. Using powerful languages makes the solutions in the GoF book look hopelessly complicated. This session shows how to solve the same problems concisely, elegantly, and with far fewer lines of code using the facilities of dynamic languages.

Introduction to JRuby by Neal Ford

This session describes JRuby, the 100% pure-Java implementation of the Ruby programming language. It covers the basics of programming with JRuby and examples of how to integrate it into existing Java projects.

Rails for JRuby by Neal Ford

This session explains all the hype surrounding Ruby on Rails, in a context familiar to Java developers. It covers convention over configuration, ActiveRecord, controllers, views, Ajax, scaffolding, testing, and deployment...on the JVM, using JRuby.

Code Metrics & Analysis for Agile Projects by Neal Ford

What does code + methodology have to do with one another? Everything! Agile projects focus on delivering working code, and tools exist to allow you to verify some quality metrics for your code. This session is a survey of tools and metrics that allow you to determine the quality of your code and strategies to "wire it" into your agile project.

Regular Expressions in Java by Neal Ford

Regular expressions should be an integral part of every developer's toolbox, but most don't realize what an important topic it is. Regular expressions have existed for decades, but many developers don't understand how to take full advantage of this powerful mechanism, either through command line tools and editors or in their development.

Agile Project Management with Mingle by Neal Ford

Mingle is an innovative project management tool with "skinnable religion", virtual card walls, highly customizable workflow, and role-based views. This talk describes its setup, use, and some implementation details of how it was created.

10 Things Every Software Architect Should Know by Richard Monson-Haefel

An effective software architect understands that every application is different and requires unique choices regarding programming language, middleware, integration, data access, user interface design, etc. Richard Monson-Haefel has distilled knowledge from his own experience and from personal interviews with the

World's best software architects to define 10 principles every software architect should know in order to be effective.

Developing Rich Internet Applications by Richard Monson-Haefel

With literally hundreds of RIA products (e.g., Adobe Flash, Nexaweb, Backbase) and open source Ajax projects (e.g. Dojo, GWT, Prototype) to choose from. Picking the right RIA technology for the job requires months of research. Richard Monson-Haefel has been researching and writing about RIA alternatives for two years and has already done the research so you don't have to.

Understanding Open Source Licensing by Richard Monson-Haefel

What does GPL, LGPL, MIT, Apache licenses, copyleft, and dual licensing mean? Richard Monson-Haefel explains both the legal and technical implications of the major open source licenses in plain English. He explains when and how you can use open source in the enterprise and in the development of software products and how to protect your organization from abusing open source licensing.

Groovy, the Blue Pill: Writing Next Generation Java Code in Groovy by Scott Davis

There are wild-eyed radicals out there telling you that Java is dead, statically-typed languages are passe, and your skills are hopelessly out-of-date. Those extremists are the same ones who don't bat an eye at throwing out years of experience to learn a new language from scratch, pushing aside a familiar IDE for a new one, and deploying to a whole new set of production servers with little regard to legacy integration. While this "burn the boats" approach to software development might sound exciting to some folks, it's giving your manager the cold shakes right now. What if I told you that there was a way that you could integrate seamlessly with your legacy Java code, continue to use your trusty IDE and stable production servers, and yet take advantage of many of the exciting new dynamic language features that those fanatics keep prattling on about? You'd probably say, "Groovy!" I would, too...

Groovy, The Red Pill: Metaprogramming, the Groovy Way to Blow a Buttoned-Down Java Developer's Mind by Scott Davis

This talk focuses on the ways that Groovy can turn a traditional Java developer's world-view upside down. We'll start by talking about how you can thumb your nose at The Man by leaving out many of the main syntactic hallmarks of Java: semicolons, parentheses, return statements, type declarations (aka Duck-typing), and the ever-present try/catch block. Then we'll look at features like operator overloading and method pointers that Groovy welcomes back into the language with open arms.

Grails for (Recovering) Struts Developers: A Groovy Alternative by Scott Davis

Struts enjoys an unprecedented marketshare in the Java web development space -- 60%-70% according to most surveys. As newer, modern web frameworks come to the scene, very little attention is paid to the real costs of migrating an existing Struts application. This talk shows you ways to mix Groovy into a legacy Struts application, dramatically reducing both the lines of code and the complexity. We'll also introduce you to Grails (a Groovy-based web framework) whose URL-mapping capabilities allow it to replace your Struts application without breaking legacy URLs.

Real World JSON by Scott Davis

JavaScript Object Notation is becoming a familiar delivery platform for Web 2.0 content. JSON gives you all of the flexibility of a RESTful web service without the hassle of trying to deal with deeply nested, complex XML in a language that is conspicuously lacking in native XML support. In this talk, we look at popular websites (like Yahoo!) that offer JSON output. We look at client-side JavaScript code that effortlessly consumes JSON in the browser. We even look at ways to easily generate JSON from Java Servlets (using JSON.org libraries) and the native support for JSON that Grails offers out of the box.

YSlow: Building Your Website for Speed by Scott Davis

How optimized is your website? YSlow, a FireFox/FireBug plugin, doesn't pull any punches. It gives any website an A, B, C, D, or F rating based on 14 individual analysis points. You'll be amazed (or depressed) at what YSlow thinks of your site. In this talk, we'll walk through these points step by step, learning what Yahoo! (the creator of this utility) does to keep its web properties running as quickly as possible.

The Busy Java Developer's Guide to Debugging by Ted Neward

Bugs? We all know your code has no bugs, but someday, you're going to find yourself tracking down a bug in somebody else's code, and that's when it's going to be helpful to have some basic ideas about

bug-tracking in your toolbox. Learn to make use of the wealth of tools that the Java Standard Platform makes available to you--tools that your IDE may not know exist, tools that you can make use of even within a production environment.

The Busy Java Developer's Guide to Monitoring by Ted Neward

Crashes? Outages? Slow response? We all know that it's never your code that causes these things, but for some reason those pesky system administrators still insist on paging you at 4AM to come in and fix those problems, anyway. For some reason, they just keep expecting you to support this thing, even after QA said it was OK!

The Busy Java Developer's Guide to Performance and Scalability by Ted Neward

Wondering why your enterprise Java app just... sucks? Trying to figure out why you can't get more than 10 concurrent users online at the same time? Looking for ways to try and spot the slowdowns and ways to fix them?

The Busy Java Developer's Guide to Concurrency (Part 1: Threads) by Ted Neward

Java's threading capabilities took a serious turn for the better with the release of Java5, thanks to the incorporation of the `java.util.concurrent` packages, a set of pre-built components for thread pooling and execution, synchronization, and more.

The Busy Java Developer's Guide to Concurrency (Part 2: Concurrency) by Ted Neward

Java's threading capabilities have been a part of the Java platform since its inception, yet for many Java developers, using Threads still remain a dark and mysterious art, and synchronization beyond the use of the "synchronized" keyword is almost unknown. **Prerequisite:** *The Busy Java Developer's Guide to Concurrency (Part 1: Threads)*

The Busy Java Developer's Guide to Hacking (on) the JDK by Ted Neward

Ever since its 1.1 release, the Java Virtual Machine steadily becomes a more and more "hackable" (configurable, pluggable, customizable, choose your own adjective here) platform for Java developers, yet few, if any, Java developers take advantage of it. Time to take the kid gloves off, crack open the platform, and see what's there. Time to play.

Know your Java? by Venkat Subramaniam

Java has been around for well over a decade now. It started out with the goal of being simple. Over the years, its picked up quite a bit of features and along comes complexity. In this presentation we will take a look at some tricky features of Java, those that can trip you over, and also look at some ways to improve your Java code.

BDD in Java and Groovy by Venkat Subramaniam

In this presentation we will take a look at what BDD is and look at tools to create them in Java and Groovy.

Design Patterns in Java and Groovy by Venkat Subramaniam

You're most likely familiar with the Gang-of-four design patterns and how to implement them in Java. However, you wouldn't want to implement those patterns in a similar way in Groovy. Furthermore, there are a number of other useful patterns that you can apply in Java and Groovy. In this presentation we'll look at two things: How to use patterns in Groovy and beyond Gang-of-four patterns in Groovy and Java.

DSL in Groovy by Venkat Subramaniam

DSL or Domain Specific Languages focus on a domain or problem at hand. They're expressive, but their restricted scope keeps them simple and small from the user point of view. However, designing them is not easy. In this presentation we will explore the features of Groovy and show how they can be used to create DSLs.

Acceptance Testing Application Behavior by Venkat Subramaniam

How do you ensure your applications meet the expectations of your key customers? In this session we will explore using the FIT tool and Behavior Driven Design tools to do exactly this.

Caring about your Code Quality by Venkat Subramaniam

We all have seen our share of bad code. We certainly have come across some good code as well. What are the characteristics of good code? How can we identify those? What practices can promote us to write and maintain more of those good quality code. This presentation will focus on this topic that has a major impact on our ability to be agile and succeed.

Design Patterns in Java and Groovy by Venkat Subramaniam

You're most likely familiar with the Gang-of-four design patterns and how to implement them in Java. However, you wouldn't want to implement those patterns in a similar way in Groovy. Furthermore, there are a number of other useful patterns that you can apply in Java and Groovy. In this presentation we'll look at two things: How to use patterns in Groovy and beyond Gang-of-four patterns in Groovy and Java.